CS-1Q HCI: Exercise Notes

C.W. Johnson,
University of Glasgow, Glasgow, G12 8RZ. Scotland.
johnson@dcs.gla.ac.uk, http://www.dcs.gla.ac.uk/~johnson

Contents

Tutorial and Lab Instructions

1 Introduction

The course notes and all other associated resources except for this handout can be found on:

`http://www.dcs.gla.ac.uk/~johnson/teaching/CS-1Q/`

The HCI lectures will take place in the first five weeks of term. The sixth week will be devoted to a short course on Professional Issues which is also covered by these notes. Table 1 provides a slightly more detailed overview of the timing of the lectures, the labs, tutorials and the hand-in deadlines. This is an ideal timetable and in practise we are unlikely to cover all of the topics listed here. I'll try to take a vote on those areas that interest people in the class.

Week	Lecture Topics	CS-1Q Lab/Tutorial	Work due
Week 1	1: Users 2: Computers	No CS-1Q Labs	
Week 2	3: Elicitation 4: Evaluation	Tutorial: Satisfaction	
Week 3	5: Distribution 6: Groups	Lab: Surfing	
Week 4	7: Desktops 8: Mobility	Tutorial: Prototyping	Analysis of web usability heuristics from previous lab. Not degree assessed.
Week 5	9: Error 10: Fun	Lab: Evaluation	
Week 6	11: Security 12: Safety	Tutorial: Trial Exam	
Week 7	Information Management (IM) starts	Lab: Professional Issues	Design and evaluation of prototype system. Degree assessed (6% of CQ-1Q)
Week 8	IM continues	IM tutorial	Professional issues essay. Degree assessed (2% of CQ-1Q)

Table 1: CS-1Q HCI Timetable

There are no extensions for any submissions. If the work is handed-in late then it should not be marked. The course handbook acts as the primary reference for this policy and if there is any doubt please refer to this first. Remember to submit a 'statement of originality' form with all degree assessed work.

2 Tutorial Week 2: User Interface Satisfaction

2.1 What is Usability and How Do We Assess It?

It's often very unclear what people mean by the term 'usability'. For example, a system that one person finds very easy to use can be frustrating and annoying for other people. This creates problems for the designers of human computer interfaces. Very often contracts are issued with terms that specify an 'improvement in usability' or that a particular design will support 'ease of use'.

Rather than talk about usability as an abstract concept, it is better to break it down into a number of more detailed ideas. We can talk about learning; how quickly can certain types of user learn to perform specified tasks? An interface can also be assessed to see whether it induced particular errors; how many 'mistakes' do users make in performing those tasks and can then learn to correct them? We can assess speed of use; how quickly can users perform those tasks given a specified amount of training? We can also assess user interface satisfaction; do users enjoy using the system and would they use it again? This is an incomplete list but it should give you some idea of the concepts that often masked by the (ab)use of the term usability. It is also important to remember that if we want to measure 'usability' in terms of these different concepts, we must also be very clear about the intended users of the system. For instance, by attending a top rated computer science department in a top rated University, you are amongst the educational elite of this great nation. Therefore, any results that might be obtained from testing the time that you take to learn how to operate a system definitely will not be applicable to society in general.

There are a number of well established techniques that can be used to evaluate an interface in term of the various concepts listed above. For example, we can measure the speed of interaction using laboratory-based tests with pre-defined tasks in the manner that is described in the lecture on evaluation. Notice that we can use evaluation techniques not only to asses new designs but also to assess the 'usability' of an existing interface. Similarly, we can assess the time taken to learn how to use an interface by measuring the length of time that it takes a novice user to feel confident in performing a particular operation. The number of errors might be assessed using cooperative evaluation techniques where the designer asks the user to operate the system. The user has to tell them if they get stuck or if they don't know what to do next. The designer can then note down any problems with the design or 'mistakes' in the user's understanding of how to operate the system.

Unfortunately, some of the components of 'usability' are less well defined and are consequently harder to evaluate. For example, how do we assess the level of satisfaction that a user experiences with a computer system. This can be a strange question for some systems. How would you express your level of satisfaction with a Python programming environment? In other contexts, satisfaction is of critical commercial importance. In particular, web-based systems often rely upon advertising revenue. This, in turn, depends on the number of 'hits' that their site receives and if users aren't satisfied with a site then they won't come back.

Questionnaires provide one means of assessing satisfaction with a user interfaces. They can ask whether users strongly agree, agree, are neutral, disagree or strongly disagree with subjective statements about the quality of a user interfaces. A range of other question formats can also be used to probe more detailed aspects of an interface, for instance, by ranking particular features of an interactive system. It is also possible to construct questions that verify particular answers. For example, if a user states that the search option is the most important aspect of a web site the questionnaire might then ask them a question about the search facilities. The response can then be analysed to see how well the user understands the facilities that they value most highly. Questions can also be structured to identify those aspects of a system that they least like.

2.2 What Do We Do Now?

This exercise should be conducted in groups of two or three. You have been commissioned by the Johnson Corporation to conduct a survey into the usability of the texting interface on mobile telephones. As part of this work, you are to develop a questionnaire that will assess the good and the bad points of the commands that are available to help write SMS texts on existing mobile phones. The first page of the questionnaire should prompt the user for information about themselves. You should ask about whether or not they possess a mobile phone and, if so, assess how often they send text messages. The second page will ask questions about their experience of actually trying to create SMS texts. You might ask about the range of features that they have used. Such features can include predictive texting systems, these applications try to guess the word that you are typing and will automatically complete a word but they can be a pain to correct when the guess is incorrect. Some systems help to reduce this problem by provide dictionaries where you can enter the names of your friends or short phrases so the system is less likely to make an incorrect guess. Other phones use special input devices, including wheels that are turned to select words from a dictionary without having to use the numeric keypad on a phone. If in doubt, take a look at some of the functions on the phones in the group. You might ask about any particularly useful aspects of this system and any potentially irritating features. The idea behind the questionnaire is to obtain information about these good and bad features that could be scaled up to thousands of users rather than having to sit and ask small groups.

Once you have developed the questionnaire you should get the members of another group to complete it. Answers should be written on a separate piece of paper. The questionnaire should be anonymous. As soon as you have done this, begin to compile your results into a graph or table that you think best explains the findings from the questionnaire. Please be as fast as you can in completing this stage.

2.2.1 Questionnaire Design (15 minutes)

The questionnaire must assess the user's knowledge of a texting system for a mobile phone. For example, do they know how to access all of the features on their phone? If users do not use all of the facilities then the questionnaire must identify the reasons why not. It should also identify those enhancements that users would most want to see from a future phone.

A variety of common formats are used in questionnaires. The simplest are the simple Boolean or 'yes/no' questions:

```
Do you like the existing system?  Yes  No
```

Multiple choice questions provide a further variation on this format where users are asked to select ONE from several different alternative responses:

```
Tick the best aspect of the existing system:
      - simple screens;
      - easy to send a basic message;
      - helpful information when things go wrong.
```

There are further variations on this where users are asked to rank the options in order of preference. This can provide much more information than a multiple choice question, however, it can be difficult to draft the instructions so that all people understand what the question is asking for. I know it sounds simple but many people find this had to understand!

Likert scales can be used. These ask people to select an item that is at some point along a scale. we have already met one of these scales (strongly agree, agree, are neutral, disagree or strongly disagree) another might be bad, good, brilliant. The problem with this is that people can disagree about the difference between 'brilliant' and 'good'. In consequence, scalar values are often used (bad = 1, good = 2, brilliant = 3), this indicates that brilliant and bad are equidistant from good. A further problem with the previous scale is that most people will tend towards the middle value. One way to avoid this problem is to use the following format:

```
                              terrible     wonderful
How helpful were the lectures?    1  2  3  4  5  6     Not applicable
```

In contrast, open ended questions can elicit a lot of opinions and additional information. Unfortunately, they are less good for providing the statistical information that the Johnson Corporation might require when making specific recommendations back to their client:

```
What is good about the existing system?
```

People also will abandon the questionnaire if they have to complete too many of these sorts of questions. They often don't have the time nor the motivation to write lots and lots. In the USA, Federal organisations must tell people how long they expect it will take them to complete a questionnaire. This is generally good practise and can reassure many potential respondents.

2.2.2 Completing the Questionnaire (15 minutes)

After developing the questionnaire, try to get the members of another group to complete it. Ideally, they should answer it one at a time without consultation. They should not need to ask any questions while they do this because the form should be self-explanatory. Hint: get people to write the answers on a separate sheet of paper to avoid writing out the questionnaire several times.

Only do this if you have at least 20 minutes remaining in the tutorial. Once you have the first two or three sets of replies, begin to collate your results. Note down any problems that the 'volunteers' had in understanding the questions or completing the form.

2.2.3 Feedback Session (15 minutes)

Some of the groups should explain the questions that they used and also present some of the results that they obtained.

Key questions: What are the ethical considerations in issuing questionnaires? Usually all data must be anonymized and approval may have to be gained from the Departmental ethics committee. This will be covered in subsequent lectures.

Many people think that questionnaires are a cheap and easy means of assessing user satisfaction. Are questionnaires ease to draft? Do users always understand the questions? Why might we distrust the answers that questionnaires elicit about usability?

2.3 After the tutorial

There is a 'standard' Questionnaire for User Interface Satisfaction. This was published in: Chin, J.P., Diehl, V.A., Norman, K.L. (1988) *Development of an Instrument Measuring User Satisfaction of the Human-Computer Interface.* ACM CHI'88 Proceedings, 213-218. You can read this paper and view an on-line version of the questionnaire from:

`http://www.acm.org/~perlman/question.html`

Hint: this may be useful for the open exercise that is associated with the HCI course.

3 Lab Week 3: Surfing and Heuristic Evaluation

3.1 What is Heuristic Evaluation?

Heuristics are rules of thumb or informal guidelines that can be used to identify potential problems in a user interface. They are generally an effective and low-cost technique because they can be applied by designers and programmers without having to recruit end-users. Of course, this creates a potential problem if the experts apply the heuristics incorrectly or if the heuristics themselves don't provide valuable guidance. Further problems arise because heuristics are often written so that they can be applied to a wide range of user interfaces. This makes it difficult to know how to apply them to any particular interface. Another consequence is that different experts can reach different conclusions about how well an interface follows a particular guideline. This lab session should provide you with an opportunity to apply Shneiderman's 'eight golden rules of interface design':

1. Strive for consistency. Designers must emphasise the patterns in their design so that users can learn to use common features when they explore unfamiliar areas of a design. For example, the `File, Edit, View` menus appear consistently on most menu based interfaces. Even if users are unfamiliar with the system they can quickly learn what these menus are used for. In web design, the icons should also be used in a consistent way. For example, the same image should always lead you back to a home page. Consistency is also important in the choice of fonts, of colours, of capitalisation and so on.

2. Enable frequent users to use shortcuts. As users become more confident with an interface, it should be possible for them to learn how to do things more quickly. This increases their expertise with a system and can provide strong subjective satisfaction at getting a task done more quickly. For instance, `CTRL-S` (pressing the Control key and the S key at the same time) will save a document on most interfaces without having to use the mouse. Similarly, pressing the `TAB` key will navigate between the fields in a form without using the mouse.

3. Offer informative feedback. Whenever a use performs an action there must be some form of visible feedback. For instance, if a button is selected then the colour of the button should change or an appropriate sound should be played so that users are confident that their action has been successful. If such feedback is not provided then there is a danger that users will repeat the action until they are happy that it has been successful. This creates problems if the action is not idempotent; an action is idempotent if there is no difference between doing it once or more than once. Printing a file is not an idempotant action because if you repeat the command you'll end up with more copies of the document. If the system does not tell you that the document is being printed then you could click 10 or 11 times to be sure - killing a tree or two.

4. Design dialogues to yield closure. Sequence of actions should have a clearly marked end-point so that users are sure that they have reached the end of a task. For example, many users will forget to save a file at the end of a session. If the system does not remind them of this before they log-off then all changes can be lost.

5. Offer error prevention and simple error handling. This is fairly straightforward. The system should help the user to avoid errors. For instance, if a user enters an incorrect value into a field of a form, feedback should be given as soon as possible rather than at the end of a long dialogue when many further mistakes may have been made. However, users should also be able to interpret the seriousness of an error message from the information provided. Can they safely continue or do they need to seek help? There must be specific advice on recovery actions.

6. Permit easy reversal of actions. Actions should if possible be reversible. This will reassure the user that they can always get back to where they started. This implies the provision of 'Undo' and history mechanisms and that these features should be easily understood.

7. Support locus of control. Users should get a sense of control - the system should be responsive to their commands, feedback should be rapid. The system should avoid long and protracted dialogues, especially if many options are not applicable to the task in hand. Surprising actions and unnecessary distractions on the interface should be avoided.

8. Reduce short-term memory load. The user should not be forced to memorise arbitrary sequences of commands or screen progressions in order to complete a task. Information should be displayed to prompt the user whenever possible so that they are not forced to remember it.

It is important to remember that these are heuristics or rules of thumb. They are not legal requirements so they can be broken, especially if the designer has a good reason for doing so. In particular, some information may be presented in an inconsistent manner if it is important to attract the user's attention to it - to make it stand out. Similarly, it may not be a good idea to offer experts a short cut to actions that could have a very harmful effect (eg, erase or reformat all of the memory on this disk). It is also important to remember that it can be very hard to apply these heuristics to all interfaces. In particular, a series of heuristics have been developed for web-based interfaces. This lab session is intended to help you apply Shneiderman's heuristics to a web site and then to elicit new heuristics that you feel should have been adopted.

3.2 What Do We Do Now?

Complete this exercise individually or in teams of two and try to keep to the time limits. If you do this you should be able to complete the work in the lab session. If not then you'll have to finish it in your own time.

3.2.1 Apply the Heuristics (60 minutes)

The Johnson Corporation have been asked to find out if Shneiderman's heuristics could be used to help design the user interface to a new Wiki. The most popular example of one of these systems is the Wikipedia project on:

```
http://en.wikipedia.org
```

There is also a general introduction to these software systems on:

```
http://pbwiki.com
```

A wiki is a type of web site that enables users both to view information and to add content. There is disagreement over where the term comes from. Some argue that is means 'What I Know Is', others have argued that it comes from a Hawai'an term for 'quick' or in a 'hurry'. This derivation is important because two key aims for a Wiki is that it must provide information to readers when they need it with very easy access AND it must be very quick for individuals to contribute information to the web site so that it will expand as part of a wider on-line community exercise.

The first task in your work for the Johnson Corporation is to apply Shneiderman's rules to an existing Wiki web site. The Wikipedia project is the best known but there are many others. You are assessing the design of the web site from the perspective of a potential new user who would like to learn more about what they can gain from a Wiki and what they might contribute.

You will only have time to examine a small area of each selected site. One way of doing this is to think of the information that a potential contributor might want to find on the site and then apply the heuristics as you attempt to locate that information. This would include information about security arrangements and how material is approved after it has been edited. Warning 1: these are live web sites and so I am trusting you not to abuse them. Warning 2: these are live web sites and so may not be available during any particular lab. If this happens then you will have to return in your own time to work on the practical.

3.2.2 Devise and Write-up New Heuristics (60 minutes)

The Johnson Corporation now require that you use the results of your investigation to draft their guidelines that a client will use during the development of a new Wiki site. The client is a multinational bank that wants to launch an innovative help service. It is considering the development of a Wiki for customers to share information about banking services. Using a Wiki will enable users to add the information that they found useful or to request help from others with similar problems. This will enable the bank to identify areas where they can improve their services and ideally will help to save money by ensuring that the official help pages are well tailored to the customers' needs.

The sorts of information to be presented can be seen by accessing existing banks' web site. The innovative part will be to create an area where customers can add their own information, for example, providing information about services and information. Most Wikis enable users to add links to other pages. For example, one customer might use the Wiki to explain the best way of changing the address associated with an account. Another user might add information to the Wiki about the pieces of identification that are needed to set up an account. Such basic information is often very hard to find on existing web sites where banks spend most of their time trying to sell new services. The key idea here is that customers actively contribute to provide this type of 'self help'.

*Hint:*If you look at the pages of an existing Wiki, you can see that the pages look very different depending on whether you are just reading information or whether you are trying to add new details. For this exercise, you are identifying heuristics that would help the bank to ensure the usability of these pages both for accessing information on the Wiki and for writing new articles. You can ensure that your guidelines are appropriate by looking at the Wikipedia pages and ensuring that your suggestions would encourage designers to use the successful features of this interface.

The client is particularly concerned that these guidelines should result in pages that can be easily navigated and that no users should become 'lost' in their web site. You may feel that Shneiderman's heuristics were sufficient. In which case, you should consider how they might need to be re-expressed so that they specifically address Wiki sites rather than all interfaces.

Most corporate guidelines consist of between 7 and 10 heuristics. These should be written up together with a brief justification of why they were selected as being particularly important. This justification should cite examples from an existing Wiki that you examined. The write-ups should be no longer than **three sides of A4** and should be submitted to your tutor at the start of next week's tutorial. This work *does not* form part of the assessed component for this course. However, failure to complete this task will leave you at a potential disadvantage both in the exam and in the open exercise; both of which will be assessed.

3.3 After the Lab Session

Jakob Nielsen pioneered usability heuristics [1]. Together with Don Norman, he founded the Nielsen-Norman group. This is one of the best known US HCI consultancies and charges for access to range of commercial usability heuristics [2] Nielsen also provides 'free' access to web-based usability heuristics through his 'Alertbox' site [3].

[1] See for example J. Nielsen, *Enhancing the Explanatory Power of Usability Heuristics Tools for Design*, ACM CHI'94, v.1 p.152-158.`www.acm.org/pubs/articles/proceedings/chi/191666/p152-nielsen/p152-nielsen.pdf`

[2] `http://www.nngroup.com`.

[3] `http://www.useit.com/alertbox/`

4 Tutorial Week 4: Prototyping

4.1 Introduction to the Assessed Exercise

The HCI open assessment is composed of two stages. In this tutorial you will begin to develop a prototype. In next week's lab you will conduct a formative evaluation of the design and identify any necessary changes to your interface. A write-up of the work should be submitted to your tutor at the start of the Lab in week 7. Please remember to include a 'statement of originality form' with your submission, as described in the CS1Q course handbook. The final submission will contribute 6% to your overall mark in CS-1Q. The open exercise in information management will contribute 6%, systems engineering will contribute 6% and a professional issues exercise will contribute 2% to your your final mark. Please consult the course guide for information on what to do if medical circumstances prevent you from completing the exercise by the deadline.

4.2 What is User Interface Prototyping?

Prototyping involves the development of a partial implementation or a mock-up of a user interface. The intention is that this will be used to drive the *formative evaluation* of design ideas. Problems can be discovered as soon as possible in the design process so that they do not emerge as major bugs just after a product goes on the market. These prototypes are often deliberately 'low cost' approximations to a final system so that designers can experiment with lots of ideas without blowing all of the development budget. One variation of this is 'throw away' prototyping that deliberately intends that most of the designs will be discarded prior to implementation. In consequence, interface designers will often develop a large number of sketches of an interface before they cut any code. These sketches can be shown to users who must then 'pretend' to work their way through each successive screen. They can point to menu items that are drawn on the sketches. The designer must then prepare another sketch of what the interface would look like then. In this way, a story-board is developed for interaction with the system. For the assessed exercise, you can use any prototyping tools that you like. These can include computer-generated mock-ups or 'pencil and paper' sketches. For the prototyping and evaluation sections of this exercise you will be working in teams of not more than two people. The associated write-up must be completed entirely on your own.

4.3 What Do We Do Now?

The Johnson Corporation were very impressed with your work on subjective satisfaction and on the heuristic evaluation of a Wiki site. This time, they want you to design an interface for the bank's Wiki that you began to consider in the previous session. Recall that this interface is intended to provide a self-help facility for the bank's customers as they try to locate information about their services.

You do NOT have to develop an implementation for the design. Rough sketches of the interface are sufficient. There are many things to think about. In particular, it is unclear whether many customers would ever be motivated to contribute to such a system. If they are interested in providing help that other might use then the interface must make it extremely easy to add a new item. However, you might also design parts of the interface to help bank employees moderate the information and check for any inaccuracies. Remember also that customers may have significant security concerns and worries over the way that any contribution to the Wiki will be handled - for example, they might be concerned that any information they provide might be traced back to them.

4.3.1 Identify User Characteristics (10 minutes)

The first stage of your design task is to identify the characteristics of your intended user market. Who will they be? What can you assume about their knowledge of information technology? In particular, how easy would it be to design an interface to add normal text but also enable contributors to create links to other pages within the banks web site and beyond.

4.3.2 Identify Tasks to be Supported (10 minutes)

You must identify the main tasks that will be supported by the system. The design must consider the contributors to the Wiki and the information consumers or readers.

4.3.3 Develop design (30 minutes)

Most of this tutorial should be spent in sketching out the initial screens that will be used to support your design. This can be done by pencil and paper sketches. Do NOT lose them. You can compare these to the existing screens that Wikipedia and other sites use to support information editing. Please ensure that each member of the group makes their own copy. This is vital because each year some people are ill or go missing and cannot attend subsequent labs and tutorials. If you do not make a copy then this may affect YOUR mark.

Hints: you will be asked to evaluate your prototype in the next session. You only need to sketch a small number of sample screens, do not attempt to develop a prototype of the complete on-line service.

5 Lab Week 5: Evaluation

5.1 Introduction to the Assessed Exercise

The HCI open assessment is composed of two stages. In last week's tutorial, you began to develop a prototype. In this week's lab, you should complete the development of your prototype and conduct an evaluation of your design. Any work that is not finished by the end of this lab MUST be completed in your own time before next week's tutorial. A write-up of the work should be submitted to your tutor at the start of the Lab in week 7. Again, remember to include a statement of originality with your work. The final submission will contribute 6% to your overall mark in CS-1Q. The open exercise in information management will contribute 6%, systems engineering will contribute 6% and a professional issues exercise will contribute 2% to your your final mark. Please consult the course guide for information on what to do if medical circumstances prevent you from completing the exercise by the deadline.

5.2 What is Evaluation?

The lectures will introduce the concepts of *formative* and *summative* evaluation. Formative evaluation helps you to develop your design ideas. As you construct a potential interface there are usually lots of ideas that you may feel unsure about. User testing provides a means of getting feedback on whether or not those ideas will work. This form of evaluation is usually conducted with the idea that it is still possible to make significant changes if the tests do not go well. In contrast, summative evaluation occurs at the end of the development cycle. It is, typically, used to demonstrate that your interface meets the clients' and users' requirements. Ideally, formative evaluation will have identified any major problems because there may be insufficient development resources left to correct any major problems that are identified through summative evaluation. Given the limited development opportunities that you have been given for your Wiki site, you should focus on the formative evaluation of your design.

In the lectures, we have identified a range of evaluation techniques. These include usage diaries, 'think alouds' or cooperative evaluation, lab based experiments, ethnomethodology. It is up to you which technique you select for the open assessment. However, you must justify your decision in the write-up that is to submitted in the lab of week 7.

5.3 What Do We Do Now?

The previous tutorial sheet asked that every member of each design team had a copy of the design. If this has been done then everyone in the lab should have an interface to evaluate even if one of their colleagues is missing. Working in your original groups, you should first complete your prototype interface.

5.3.1 Select the Evaluation Technique (10 minutes)

Next, you should agree upon the evaluation techniques that you will use. One means of doing this is to write down the strengths and weaknesses of each of the approaches that might be considered. You should match the strengths of each technique against the key features or design decisions that you would like to find out about during the evaluation. Remember that there a lot of obvious problems here. Firstly, your prototype is likely to be based on a pencil and paper sketch and so will provide little of the 'look and feel' of a real on-line service. It will, therefore, be difficult to simulate the effects of retrieval delays or of mis-keyed input. Secondly, other people will be working on the same exercise. This may mean that any feedback you get from people will not reflect the views and experience of more ‘naive’ users who have little experience of Wiki’s.

5.3.2 Design the Evaluation (20 minutes)

After having documented the reasons for and against different evaluation techniques, it is time to start planning the chosen means of validating your design. Exactly what you have to do will depend on the technique that you have chosen.

For most evaluation techniques, it will be necessary to decide on the ordering of the tasks that will be assessed. This is important because there can be learning and fatigue effects. Users will do better on subsequent tasks as they learn how to navigate the information in your system. Conversely, users may do worse on subsequent tasks as they become tired by the effort of concentrating over a prolonged period of time. You must also decide how many users you will ask to participate in the evaluation. You must consider whether it is sufficient simply to test your interface with other people of the same age, gender and educational background.

Ideally, you should design evaluations to consider different user roles. In other words, yuo should assess the usability of the system for people who are looking for information and also for users who want to contribute new information to the Wiki site.

5.3.3 Run the Evaluation (60 minutes)

You should then implement the evaluation plan that you have developed. It is normal to make changes to such a plan after the first of second test. Often users hit problems that you had never anticipated. Users are often unsure how to get start using a new site. It may, therefore, be necessary to make small changes to the web design as you go along. It is important that these changes are noted by each member of the team and they should be documented in the write-up, see below.

As before, the precise nature of the evaluation depends on the technique that you have chosen. If you are using a ‘think aloud’ technique then you should record every situation in which the user is uncertain or in which they make what appears to be a ‘mistake’. This might include problems when they do not know which link to follow to find particular items of information. It might also include problems in using search facilities or in tracing the way in which information has been added to the Wiki. The evaluation should consider the range of editing facilities that will be provided to anyone editing a page. You should analyse any problems in the write-up and propose design alternatives.

5.3.4 Make Notes on the Evaluation (30 minutes)

At the end of the lab, it is important that you make sufficient notes for you to be able to complete the write-up that is associated with this open exercise. The content of this report is described below. In particular, if one member of the group has been responsible for taking notes during the evaluation then it is vital that a copy of those notes be made before you leave the lab. Failure to do this could affect the report and the marks of the other member of the group.

5.4 After the Tutorial

You should complete the evaluation if you were unable to do this in the lab. You must then complete the write-up associated with this exercise. The Johnson Corporation have requested a project report on your evaluation and design of the Bank's Wiki site. The report should include the following sections:

1. *Requirements Analysis* (4 marks).
   You must document your analysis of the potential users of the service. You must also explain the perceptual, cognitive and physiological constraints that might affect the operation of the interface. You must describe the tasks that you have chosen to support and justify your selection.

2. *Design Summary* (6 marks).
   You should describe the interface that you have designed. One of the best means of doing this is through the use of a story-board. This is done by sketching the initial display. You should then describe the user's input and next to it show how the display would change for each step of their task. The story board should also show any error messages or help facilities that have been included within the design.

3. *Evaluation* (6 marks).
   You should describe the formative evaluation technique that you have used to assess your interface design. You must justify your choice of technique and also explain why you decided not to use other techniques. You should present the results of your evaluation and describe any problems or limitations that complicate the interpretation of your results.

4. *Further Development* (4 marks).
   You should describe any changes that you would make to your design after the evaluation. You should also briefly outline recommendations for the future development of interactive services by the Johnson Corporation.

5. *Presentation* (10 marks)
   The document should be in the style of a business report and must not exceed 5 sides of A4. This report should be submitted to you tutor at the start of the Lab in Week 7. 10 marks will be allocated for the quality of the prose and the overall presentation of the report.

The final mark will be out of 30 and this will contribute 6% to the CS-1Q course.

6 Tutorial Week 6: Practice Exam

6.1 Introduction to the Class Test

The CS-1Q class test will take place in January (week 13). It will consist entirely of Information Management and HCI material because the Systems Engineering material is covered in the second half of the year. However, the final CS-1Q exam will also cover this later material in addition to the topics that are covered in the class test. Your overall grade is determined as follows:

- 70% contribution from the degree exam;
- 20% contribution from the HCI, information management, systems engineering and professional issues open assessments;
- 10% contribution from the class test.

The purpose of this exercise is to give you a brief introduction to the style of questions that you will meet in the HCI component of the CS-1Q class test. It should also provide some experience in the format of questions that will be included in the HCI component of the degree exam.

6.2 What Do I Do Now?

Working on your own you should attempt to answer the following question. This is last year's exam question but will give you a good idea of what you will get in the class test You should work under examination conditions. Your answers must be legible and should clearly address the questions. Avoid waffle and try not to provide the answer to a question that does not appear on the paper . You should consult the course notes if you need to. After 40 minutes you should stop and pass your answer to another person in the class. They will then attempt to mark your script. There will then be an opportunity to discuss your solutions and your marking of your partner's script.

Note that because this question is based directly on last year's exam, we may not have covered all of the material in the course by this tutorial. You should, therefore, spend a little time before this week's session covering any additional topics that might be necessary to answer the following.

6.2.1 Answer the Sample Questions (40 minutes)

a) Please provide BRIEF answers to the following questions:

i) Name two differences between short and long term memory.
ii) Why is functional ageing important for the design of interactive systems?
iii) When would you use a serif font?
iv) Describe the relationship between challenge and control in Csikszentmihalyi's 'flow' theory of game playing.
v) What are the benefits of observational rather than lab based evaluations?

[2 marks per answer, 10 marks in total]

b) The screen-shot in Figure 1 is taken from the Microsoft XP operating system. As can be seen, it helps users to manage the security settings for their computer[4].
Briefly describe the main features that are you feel will support the user of this operating system. (Hint: you do not need to describe the detailed operation of any of the particular services, you do need to comment of the interface design, choice of widgets etc).

[6 marks]

c) You have been asked to design a commercial web-site. Users will be able to browse or search for music and then download it to their hard disk and any associated devices, such as MP3 players. Briefly explain how you would identify the potential end-users of such a service and then explain how you would conduct a summative evaluation for these users once the system had been built.

[9 marks]

6.2.2 Mark the Sample Questions (10 minutes)

As mentioned, you should pass your answers to another member of the tutorial group. Ideally, this should be someone that you do not know and have not spoken to before. This will avoid personal feelings biasing the marking process! Everyone should mark the scripts against what they feel the correct answer should be. Marks should be deducted if the answers are in note form, if they do not address the question as it is written or if they handwriting is illegible. I have decided not to distribute sample solutions until after the tutorial to avoid the temptation for people to look at the answers as they complete the questions. I will provide sample solutions after the week's tutorials have been completed. It is tempting to spend lots of time considering an appropriate mark for each answer. The reality is, however, that the markers of many examinations must work to tightly specified deadlines if they are to provide results back in time for the examiners' meetings. In consequence, you should spend no more than ten minutes in reaching your decisions. Please use the marking scheme that is printed next to each question.

6.2.3 Feedback (5 minutes)

At the end of the marking session, give the script back to the person who write it. Take a look at your own marks. Next go through each solution working with your partner. Describe the reasons why you awarded a particular mark. Were there any parts of their answer that you did not understand? Did you have any problems with the presentation or hand-writing? Finally, if you have time consider ways in which you might combine elements of each person's solution to derive a better answer.

6.3 After the Tutorial

There are several sample exam papers with solutions on the course web site. The solutions for this question are included there too. You should attempt to answer the questions and check your version against the sample solutions before the class test. This completes the HCI component of the CS-1Q course. If you have any questions about this material, either now or during your revision then please email me.

[4] Apologies that this is not included on the same page as the question but I wanted it to be reproduced to full size

7 Lab Week 7: Professional Issues

7.1 Introduction to the Assessed Exercise

The professional issues lectures will take place in week 6. The notes will be placed on-line so that anyone who wants to can make an early start on their essay. I would advise people not to submit their solution until after the lectures. The lab session in week 7 is left free for you to work on your solutions. Tutors will, however, be present to provide some feedback on any ideas that you may have for a potential solution. A write-up of the work should be handed to your tutor during the tutorial in week 8. The final submission will contribute 2% to your overall mark in CS-1Q. The open exercise in information management will contribute 6%, systems engineering will contribute 6% and the HCI exercise contributed 6% to your your final mark. Again, include a statement of originality with your submitted work. Please consult the course guide for information on what to do if medical circumstances prevent you from completing the exercise by the deadline.

7.2 The Questions

Warning - these questions MUST NOT be answered based simply on general knowledge. You should base your answers on significant additional research. You should write an essay that addresses ONE of the following questions.

1. Most security problems stem from the end-users' complacency. Discuss.
2. The development of safety-critical software depends more on the skill of the programmer than the choice of the programming language. Discuss.
3. The growth of the Internet has done greater harm in undermining civil liberties than it has done good by promoting free speech. Discuss.

7.3 Further Guidance

The following marking scheme will be used:

- Legibility, use of references and sources - 5 marks;
- Quality of argument - 5 marks;

Your essay must not exceed five sides of A4. It must begin with an abstract of not more than 100 words. It should also have a concluding section that clearly summarises the main points of your argument. All references and sources of information must be acknowledged in a reference section.

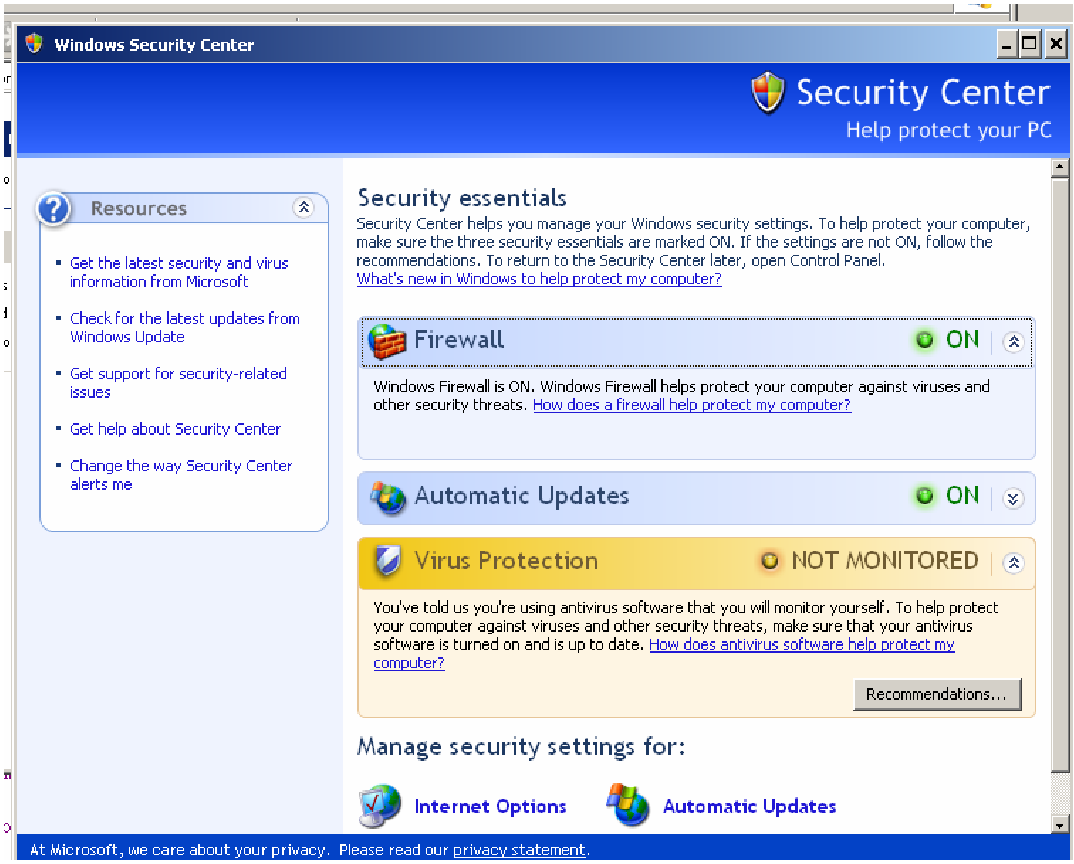
You must look at one of the on-line sources of help on how to format your references. For example:

`http://library.duke.edu/research/citing/`

This section need not be included within the page limit mentioned above. Sufficient detail must be provided for each article so that is it possible for the marker to unambiguously identify the source of the information. The use of web pages is permitted provided that some consideration is given to whether or not the information comes from a 'reliable source'. The use of web pages as a reference source is NOT sufficient to obtain a high mark. In other words, it is expected that all answers will draw upon other forms of published work including journals and books.

7.4 After the Lab

The lectures will switch in week 7 to the Information Management material. Please hand-in the Professional Issues essay during the tutorial in week 8



- Get the latest security and virus information from Microsoft
- Check for the latest updates from Windows Update
- Get support for security-related issues
- Get help about Security Center
- Change the way Security Center alerts me

Security essentials

Security Center helps you manage your Windows security settings. To help protect your computer, make sure the three security essentials are marked ON. If the settings are not ON, follow the recommendations. To return to the Security Center later, open Control Panel.
What's new in Windows to help protect my computer?

Firewall ● ON

Windows Firewall is ON. Windows Firewall helps protect your computer against viruses and other security threats. How does a firewall help protect my computer?

Automatic Updates ● ON

Virus Protection ● NOT MONITORED

You've told us you're using antivirus software that you will monitor yourself. To help protect your computer against viruses and other security threats, make sure that your antivirus software is turned on and is up to date. How does antivirus software help protect my computer?

Manage security settings for:

At Microsoft, we care about your privacy. Please read our privacy statement.